A Mixture-of-Experts Framework for Adaptive Kalman Filtering

Wassim S. Chaer, Robert H. Bishop, *Member, IEEE*, and Joydeep Ghosh

Abstract— This paper proposes a modular and flexible approach to adaptive Kalman filtering using the framework of a mixture-of-experts regulated by a gating network. Each expert is a Kalman filter modeled with a different realization of the unknown system parameters such as process and measurement noise. The gating network performs on-line adaptation of the weights given to individual filter estimates based on performance. This scheme compares very favorably with the classical Magill filter bank, which is based on a Bayesian technique, in terms of i) estimation accuracy, ii) quicker response to changing environments, and iii) numerical stability and computational demands. The proposed filter bank is further enhanced by periodically using a search algorithm in a feedback loop. Two search algorithms are considered. The first algorithm uses a recursive quadratic programming approach which extremizes a modified maximum likelihood function to update the parameters of the best performing filter in the bank. This particular approach to parameter adaptation allows a real-time implementation. The second algorithm uses a genetic algorithm to search for the parameter vector and is suited for post-processed data type applications. The workings and power of the overall filter bank and the suggested adaptation schemes are illustrated by a number of examples.

I. INTRODUCTION

A KALMAN filter requires an exact knowledge of the model parameters for optimal performance. A well-known problem in the implementation of the Kalman filter is the process of selecting the filter parameters. Incorrect modeling in the filter can lead to larger estimation errors or divergence [1], [2]. Many approaches to the problem of adaptive Kalman filtering (where the adaptation is with respect to the filter parameters) have been considered in the literature. Mehra [3] groups and discusses the main methods: correlation [4], [5], covariance matching [6]–[8], Bayesian [9]–[11], and maximum likelihood [12]–[14]. Of particular interest in this paper is the Kalman filter bank approach, first proposed by Magill [9] and subsequently categorized as a Bayesian method by Mehra [3].

In the past two decades, several algorithms have emerged to handle unknown parameters [15]–[18]. In particular, the interacting multiple model (IMM) algorithm [18] has received a lot of attention [19]–[23]. IMM provides an improvement over Magill's scheme and general pseudo Bayes (GPB) methods [15]–[17]. The IMM algorithm assumes that i) the unknown parameter vector takes a finite number of values according to a known transition probability matrix, and ii) requires a filter for the unknown parameter vector. Also, IMM is less computationally demanding than GPB but more than Magill's scheme.

This paper proposes an adaptive filter bank which can i) select "on-line" the best filter realization using a learning network, ii) respond rapidly to changing environments by being able to modify its selection based on the most recent data, iii) adapt individual filters to better match incoming data, and iv) avoid numerical instabilities and computational demands of current techniques for the Magill filter bank. Unlike the Magill scheme, the mixture-of-experts framework does not assume that the optimal filter is included in the bank. In practice, this allows the proposed scheme to produce good results with smaller filter banks. Moreover, it does not require the assumptions of IMM mentioned above, and is computationally faster.

Fig. 1 illustrates the overall adaptive filtering structure consisting of a *filter bank* and a *learning network* in a forward loop, together with a *search algorithm* in a feedback loop. The forward loop can be viewed as a multiple hypothesis estimation algorithm, wherein different realizations of the unknown (or uncertain) system parameters are coded into each individual filter in the bank and the learning network decides which realization provides the best estimate given the available input data. The feedback loop, which contains the search algorithm, is used to periodically update the various filters in the filter bank utilizing the information learned about the system in the forward loop. Note that the adaptive filtering approach suggested by Fig. 1 is a general structure that can be used for other filter models too.

The filter bank is composed of a finite number of filters running in parallel, each modeled with a different realization of the unknown parameters. In a general case, the unknown parameter vector can include different process models as well as different process and measurement noise. In many cases, the uncertainties in modeling the process and measurement are accounted for via the process and measurement noise. A recursive weighting function (i.e., the learning function) is then used to weight the outputs of the filters. The weight factors are computed as input signals (or measurements) are processed by the filter bank. The bank of filters *learns* which filter is performing best by examining a given performance

W. S. Chaer and R. H. Bishop are with the Department of Aerospace Engineering and Engineering Mechanics, University of Texas, Austin, TX 78712 USA.

J. Ghosh is with the Department of Electrical and Computer Engineering, University of Texas at Austin, TX 78712 USA.
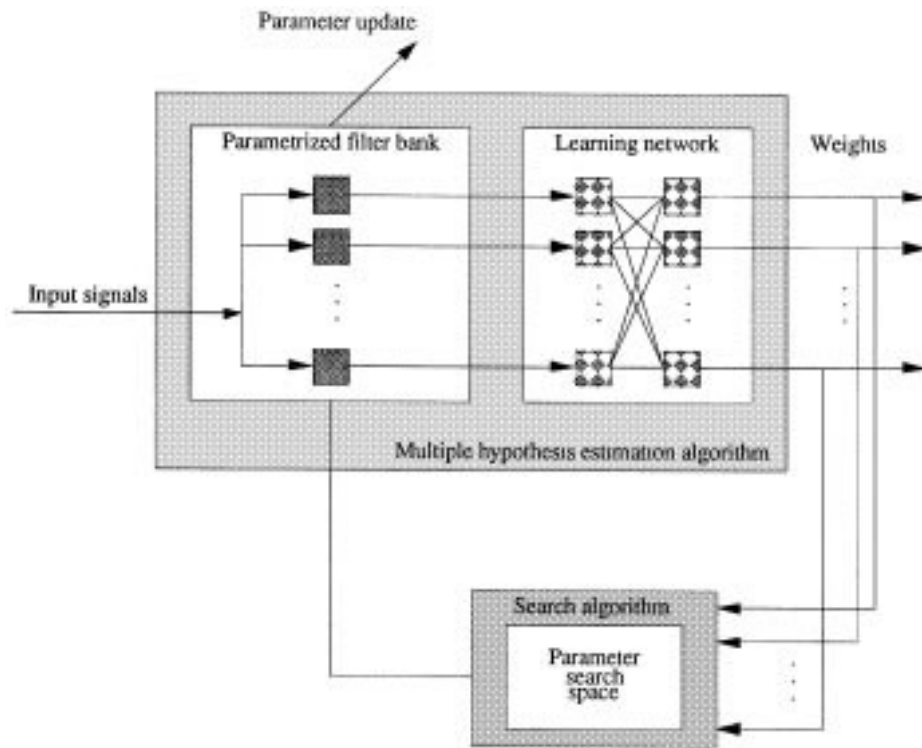
Fig. 1. Generalized adaptive filtering structure.

measure. When all input processing is completed, the highest weight factor will correspond to the best performing filter. The weight information is also used by a search algorithm to *update* the filter bank. The filter bank update corresponds to intelligently speculating on different realizations of the unknown parameters.

We introduce a new learning network to mediate among the competing Kalman filters. This network is akin to the gating network used in mixture of experts systems recently popularized in the neural network literature [24]–[26]. The different Kalman filters are the individual experts in our system. As measurements are processed, the gating network adaptively assigns weights to the individual filters in proportion to their performance. For comparison, the conventional Bayesian method proposed by Magill [9] is also implemented since it exists in actual systems today [27]–[30]. The Magill scheme is capable of converging to the correct hypothesis filter, or to the best performing hypothesis filter (when the bank does not contain all the possible realizations). It is observed that the gating network addresses several of the problems commonly encountered with Magill's Bayesian scheme, namely, numerical underflows and relatively long switching times in the presence of changing parameters.

For filter banks that are not adaptive in the parameters, i.e., if filters in the bank are fixed, a very large number of filters may be needed to cover all possible realizations. For example, if we have three unknown parameters each of which has 10 possible realizations, a filter bank of size 1000 is required! We tackle this problem by making the bank adaptive in the parameters. This is achieved by including a search algorithm in a feedback loop to periodically update the filter bank. This enables one to have a smaller filter bank and still get good results.

Two search algorithms are studied: recursive quadratic programming (RQP) [31] and genetic algorithms (GA's) [32], [33]. The first approach to parameter adaptation using RQP extremizes a constrained nonlinear program. Hence, by using a nonlinear programming optimization approach for the search function, applying the updates periodically (whenever the best performing filter is identified) and updating only the best performing filter in the bank, a real-time implementation can be achieved. This requires that the optimization computations occur in parallel with the normal filter bank operations. On the other hand, a GA uses the fitness of the high performance filters to direct the search for better filter model parameters. While the use of a GA increases the complexity of the adaptive filtering problem by requiring a larger filter bank, the resulting scheme becomes more natural to a nonreal time setting, such as in a filter tuning procedure or when measurement data are postprocessed. This follows from the fact that the GA will update the *entire* filter bank at one time. Then, all the measurement data is processed again, and this process is repeated until certain convergence criteria are satisfied. Etter *et al.* [34] used a similar structure to adaptively update the coefficients of an infinite impulse response (IIR) filter.

The remainder of the paper is organized as follows. Section II describes the Magill filter bank and the modeling problem encountered in the Kalman filter. Section III introduces our proposed approach to adaptive Kalman filtering using the concept of a mixture-of-experts regulated by a gating network. Section IV includes the derivation of the constrained nonlinear program whose solution is found by using a RQP

approach. The solution is used to adapt a particular filter in the bank. Section V consists of a simple tracking example where a number of experiments are conducted to illustrate the properties of the proposed scheme. Section VI considers a more realistic problem: processing satellite radar altimetry measurements. The obtained results for this problem show an improvement in estimate accuracy. Section VII describes an alternate approach for parameter adaptation using a GA. This particular approach is suited for a nonreal time setting. In Section VIII, we re-examine the simple example of Section V. A GA is used to perform the parameter adaptation task by post-processing a collected set of measurements. Concluding remarks are made in Section IX.

II. THE MAGILL FILTER BANK

The performance of the suggested adaptive filter structure is investigated by considering the modeling problem in the Kalman filter. In particular, we examine the well-known problem of determining the unknown system noise covariance matrix and the measurement noise covariance matrix.

Consider a physical system represented by

$$\mathbf{x}_{k+1} = \mathbf{\Phi}_{k+1,k}\mathbf{x}_k + \mathbf{w}_k,$$
$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k, \tag{1}$$

where $\mathbf{x}_k$ is the n-dimensional state vector, $\mathbf{\Phi}_{k+1,k}$ is the state transition matrix, $\mathbf{z}_k$ is the m-dimensional measurement vector, and $\mathbf{H}_k$ is the observation mapping matrix. The $\mathbf{w}_k$ and $\mathbf{v}_k$ noise vectors are assumed to be zero-mean, white sequences with

$$E\big[\mathbf{w}_k\mathbf{w}_i^T\big] = \mathbf{Q}_k\delta_{ki} \quad\text{and}\quad E\big[\mathbf{v}_k\mathbf{v}_i^T\big] = \mathbf{R}_k\delta_{ki}.$$

We also assume that process and measurement noise are uncorrelated.

In many cases, $\mathbf{Q}_k$ and $\mathbf{R}_k$ are unknown or known approximately. Usually, $\mathbf{Q}_k$ and $\mathbf{R}_k$ are modeled to be constant over a sequence of measurements representing a local operating regime. The elements of $\mathbf{Q}_k$ and $\mathbf{R}_k$ however can vary with time across operating regimes.

The estimate of the state can be obtained sequentially with the Kalman filter [35], [36]. The state estimate and error covariance matrix are propagated between measurements

$$\hat{\mathbf{x}}_{k+1}^- = \mathbf{\Phi}_{k+1,k}\hat{\mathbf{x}}_k^+, \tag{2}$$
$$\mathbf{P}_{k+1}^- = \mathbf{\Phi}_{k+1,k}\mathbf{P}_k^+\mathbf{\Phi}_{k+1,k}^T + \mathbf{Q}_k \tag{3}$$

where "$^\wedge$" denotes estimated state, "$+$" denotes after measurement incorporation, and "$-$" prior to measurement incorporation. The updated state estimate and error covariance are given by

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_k^-) \tag{4}$$
$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^- \tag{5}$$

where the Kalman gain $\mathbf{K}_k$ is

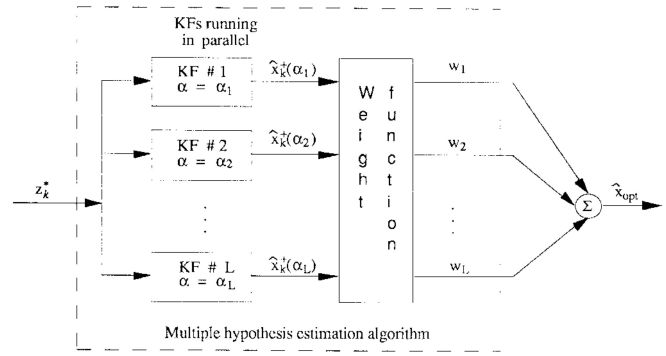$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}_k^T\mathbf{W}_k^{-1} \tag{6}$$

and

Fig. 2. The Magill filter bank and weight function.

$$\mathbf{W}_k = \mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k.$$

Equations (2)–(6) represent the standard recursive Kalman filter. The optimal Kalman filter requires an accurate knowledge of the process noise covariance matrix $\mathbf{Q}_k$, the measurement noise covariance matrix $\mathbf{R}_k$, and the parameters of $\mathbf{H}_k$ and $\mathbf{\Phi}_{k+1,k}$. In real applications, $\mathbf{Q}_k$ and $\mathbf{R}_k$ are often unknown or known approximately. This has prompted the use of a filter bank, shown in Fig. 2, which consists of L Kalman filters running in parallel, each operating with different values of the unknown parameters. The parameter vector is denoted by α_i for the ith filter in the bank, $\hat{\mathbf{x}}_k^+(\alpha_i)$ represents the output of the ith filter, and $\mathbf{z}_k^\star$ represents a sequence of k inputs $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k$. In our case, the parameter vector α_i consists of the unknown system parameters, which in this case is composed of the elements of $\mathbf{Q}_k$ and $\mathbf{R}_k$.

In general, if α is the unknown parameter vector, then the optimal estimate, $\hat{\mathbf{x}}_{\mathrm{opt}}$, can be shown to be a weighted sum of Kalman filter estimates [9], [37]

$$\hat{\mathbf{x}}_{\mathrm{opt}} = \sum_{i=1}^L \hat{\mathbf{x}}_k^+(\alpha_i)p\big(\alpha_i \mid \mathbf{z}_k^\star\big)$$

where L is the size of the Kalman filter bank. The weight factors $w_i = p(\alpha_i \mid \mathbf{z}_k^\star)$ are computed as the measurements are processed.

In the classical approach of Magill [9], the unknown parameter vector is assumed to have a finite number of possible realizations. Hence, the conditional densities for each filter can be shown to be computed recursively via

$$p\big(\mathbf{z}_k^\star \mid \alpha_i\big) = \left[\frac{1}{\sqrt{2\pi|\mathbf{W}_k|}}e^{-\frac{1}{2}\mathbf{r}_k^T\mathbf{W}_k^{-1}\mathbf{r}_k}\right]p\big(\mathbf{z}_{k-1}^\star \mid \alpha_i\big) \tag{7}$$

where

$$\mathbf{r}_k = \mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_k^-$$

is the measurement residual. Using Baye's rule, the weight factors $p(\alpha_i \mid \mathbf{z}_k^\star)$ are computed as

$$p\big(\alpha_i \mid \mathbf{z}_k^\star\big) = \frac{p\big(\mathbf{z}_k^\star \mid \alpha_i\big)p(\alpha_i)}{\sum_{j=1}^L p\big(\mathbf{z}_k^\star \mid \alpha_j\big)p(\alpha_j)} \tag{8}$$

where the distribution $p(\alpha_i)$ is assumed known. In general, $p(\alpha_i)$ is unknown, hence a uniform distribution is assumed.
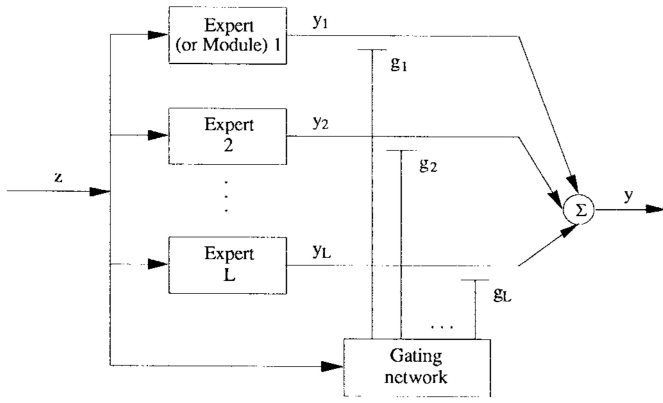
Fig. 3. A modular "mixture-of-experts" network.

Fig. 4. Gating network.

When using the filter bank and learning network as a hypothesis tester, we are not interested in the optimal estimate $\hat{\mathbf{x}}_{\text{opt}}$, but rather in the *a posteriori* probabilities $p(\alpha_i \mid \mathbf{z}_k^\star)$ of each filter.

The weight factors $w_1, w_2, \ldots, w_L$ are scalar quantities. They are computed recursively whenever an input is processed by the filter bank. A weight is maximized/minimized according to performance over the entire sequence of inputs. That is, the filter with the highest weight represents the filter with the "best" parameter vector, while the smallest weight corresponds to the filter with the "worst" parameter vector. Of course, unless the bank contains all the possible realizations of the parameter vector, the best performing filter will not necessarily be the optimal filter.

III. REGULATING MIXTURES-OF-EXPERTS

The Magill filter bank can be viewed as a modular network [24]–[26] with a generic structure as shown in Fig. 3. Such a single-level network consists of L modules called expert networks, and an integrating unit called a gating network. The input is the m-dimensional vector $\mathbf{z}$. The output of the ith expert is an n-dimensional vector $\mathbf{y}_i$. The output of the complete modular network, $\mathbf{y}$, is obtained as a weighted sum of the individual expert outputs, with the weight functions $g_1, g_2, \ldots, g_L$ supplied by the gating network.

The use of modular networks has been particularly popularized in recent neural literature, where their advantages in speed, accuracy or robustness is well documented for problems involving nonlinear regression [38], [39] and classification [40]. A modular network typically combines supervised and unsupervised learning. Supervised learning is involved during training where the desired responses, $\mathbf{d}(\mathbf{z}(\mathbf{t}))$ are specified with the input vector $\mathbf{z}(t)$. The expert networks compete to obtain the desired response. The gating network arbitrates among the different expert networks in an unsupervised fashion, by assigning higher values of the weight g_i to networks that are expected to be closer to the desired response for a given input. In particular, if the weights $g_1, g_2, \ldots, g_L$ satisfy

$$0 \leq g_i \leq 1, \quad \forall i = 1, 2, \ldots, L \tag{9}$$

and

$$\sum_{i=1}^{L} g_i = 1 \tag{10}$$
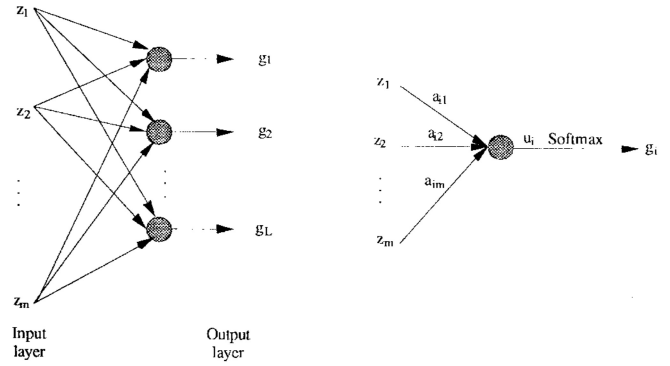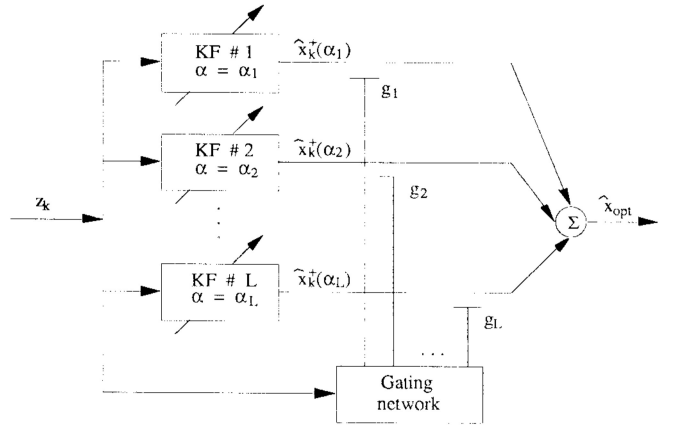

Fig. 5. Adaptive Kalman filter bank.

the weights can be interpreted as *a priori* probabilities for the corresponding experts for the given input. A sound way of achieving (9) and (10) is by defining g_i using [41]

$$g_i = \frac{e^{u_i}}{\sum_{j=1}^{L} e^{u_j}} \tag{11}$$

where

$$u_i = \mathbf{z}^T \mathbf{a}_i \tag{12}$$

and $\mathbf{a}_i$ is the weight vector of cell i, representing the ith expert in the modular network. The transformation in (11) is called *softmax* [42]. Fig. 4 illustrates the gating network used in this paper. While more powerful nonlinear maps such as a multilayered perceptron could have been used for this network, the simple structure of (11) simplifies the update equations of the synaptic weights, while still yielding an overall modular structure of sufficient power.

In this paper, each module or expert is a particular Kalman filter realization, and different modules are in competition for being selected as the most likely model. With this key insight, several results available in the theory of "mixtures of experts" networks can be gainfully applied. As shown in Fig. 5, in this paper the gating network is used as a mediator among the different Kalman filters rather than its original function of being a mediator among the different expert (neural) networks.

The Gaussian distribution of the measurement vector $\mathbf{z}_k$ of the ith filter is given by

$$f(\mathbf{z}_k \mid \alpha_i) = \frac{1}{\sqrt{2\pi |\mathbf{W}_k|}} e^{-\frac{1}{2}\mathbf{r}_k^T \mathbf{W}_k^{-1} \mathbf{r}_k}. \tag{13}$$

The probability distribution of the bank can be treated as a linear combination of the L different distributions. Then, we can write

$$f(\mathbf{z}_k) = \sum_{i=1}^{L} f(\mathbf{z}_k \mid \alpha_i) g_i. \tag{14}$$

Equation (14) can be viewed as a likelihood function. It is better to work with the natural logarithm of $f(\mathbf{z}_k)$ rather than $f(\mathbf{z}_k)$ itself. We will define the log-likelihood function as

$$l = \ln f(\mathbf{z}_k) = \ln \sum_{i=1}^{L} f(\mathbf{z}_k \mid \alpha_i) g_i \tag{15}$$

and h_i, the *a posteriori* probability associated with the ith Kalman filter, as

$$h_i = \frac{f(\mathbf{z}_k | \alpha_i) g_i}{\sum_{j=1}^{L} f(\mathbf{z}_k | \alpha_j) g_j}. \tag{16}$$

Substituting (11) in (15), we obtain

$$l = \ln \sum_{i=1}^{L} \frac{1}{\sqrt{2\pi|\mathbf{W}_k|}} e^{-\frac{1}{2}\mathbf{r}_k^T \mathbf{W}_k^{-1} \mathbf{r}_k} e^{u_i} - \ln \sum_{i=1}^{L} e^{u_i}. \tag{17}$$

To derive the update equations for the synaptic weights of the gating network, we start by computing the partial derivative of l with respect to u_i. This yields

$$\frac{\partial l}{\partial u_i} = h_i - g_i. \tag{18}$$

The sensitivity vector of the log-likelihood function with respect to the synaptic weight vector is found by applying the chain rule

$$\frac{\partial l}{\partial \mathbf{a}_i} = \frac{\partial l}{\partial u_i} \frac{\partial u_i}{\partial \mathbf{a}_i} = (h_i - g_i)\mathbf{z}_k. \tag{19}$$

Using an instantaneous gradient ascent procedure which seeks to maximize (15), the synaptic weight vector $\mathbf{a}_i$ is updated as

$$\mathbf{a}_i \leftarrow \mathbf{a}_i + \eta \frac{\partial l}{\partial \mathbf{a}_i} = \mathbf{a}_i + \eta(h_i - g_i)\mathbf{z}_k \tag{20}$$

where η is a learning-rate parameter. The update at every time step attempts to align the ith weight vector $\mathbf{a}_i$ with the input vector $\mathbf{z}_k$. This justifies the earlier inner product definition of u_i in (12). Essentially, $\mathbf{a}_i$ indicates the part of the input space where the ith filter assumes importance, since the inner product operation results in higher values of g_i for inputs $\mathbf{z}_k$ that are more aligned with $\mathbf{a}_i$.

Note that the gating network does not learn in a recursive multiplicative manner like the Magill filter bank weighting function. In the Magill scheme, this is reflected in the computation of the conditional densities for each filter as shown in (7). This is due to the assumption that all possible realizations of the unknown parameter vector (including the optimal) are in the bank. Such an assumption is not made when using the gating network. Rather, the included realizations are looked upon as candidates (not necessarily optimal). The learning done by the gating network is achieved recursively in an *additive* manner and at a learning-rate η. Typically, the gating network output for the best performing filter quickly

reaches a value close to unity. More remarkably, the gating network readily switches to another filter in the bank when performance of the current "best" filter degrades due to a change in the operational regime. The switch occurs rather fast due to the gating network property of geometrically decreasing the impact of past measurements. In other words, the gating network has a fading memory. In contrast, the Magill filter bank never forgets. It gives as much weight for early measurements as later ones. This is a direct result of the way the conditional densities are computed. This will lead to a longer time for the switching process to begin. Another advantage when using the gating network is numerical stability. In Magill's approach, a numerical underflow problem is often encountered with the computation of the conditional densities [37]. This results from multiplying exponential terms repeatedly [see (7)]. This problem does not exist when using a gating network. This is due to the different approach to learning by the gating network.

IV. PARAMETER ADAPTATION USING RQP

Besides changing the weighting of different filters, it is also desirable to adapt the parameters of at least one filter. In our current implementation, an adaptation algorithm is applied periodically or whenever a certain threshold value is reached by one of the gating weights $g_1, g_2, \ldots, g_L$. The adaptation scheme is applied only to the best performing filter in the bank. It changes the parameter vector α of that filter so as to maximize (13) for the processed measurement set observed since the previous adaptation. Assuming the measurement set is composed of p measurements, denoted by $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_p$, we have

$$\begin{aligned}
\max_{\alpha} &\, f(\mathbf{z}_p, \mathbf{z}_{p-1}, \ldots, \mathbf{z}_1 \mid \alpha) \\
&= f(\mathbf{z}_p, \mathbf{z}_{p-1}, \ldots, \mathbf{z}_2 \mid \mathbf{z}_1, \alpha) f(\mathbf{z}_1 \mid \alpha) \\
&= f(\mathbf{z}_p, \mathbf{z}_{p-1}, \ldots, \mathbf{z}_3 \mid \mathbf{z}_2, \mathbf{z}_1, \alpha) f(\mathbf{z}_2 | \mathbf{z}_1, \alpha) f(\mathbf{z}_1 | \alpha) \\
&\qquad\qquad\qquad \vdots \\
&= f(\mathbf{z}_p \mid \mathbf{z}_{p-1}, \ldots, \mathbf{z}_1, \alpha) f(\mathbf{z}_{p-1} \mid \mathbf{z}_{p-2}, \ldots, \mathbf{z}_1, \alpha), \ldots, \\
&\quad f(\mathbf{z}_2 \mid \mathbf{z}_1, \alpha) f(\mathbf{z}_1 \mid \alpha).
\end{aligned} \tag{21}$$

Recognizing that we can maximize the natural logarithm of (21) and writing our optimization problem as a minimization, we obtain

$$\min_{\alpha} J = -\sum_{i=1}^{p} \ln f(\mathbf{z}_i \mid \alpha) = \frac{1}{2} \sum_{i=1}^{p} \left\{ \mathbf{r}_i^T \mathbf{W}_i^{-1} \mathbf{r}_i + \ln 2\pi|\mathbf{W}_i| \right\} \tag{22}$$

subject to

$$\begin{aligned}
0 &\leq \mathbf{C}_{ql} \leq \mathbf{Q}_k \leq \mathbf{C}_{qu}, \\
0 &< \mathbf{C}_{rl} \leq \mathbf{R}_k \leq \mathbf{C}_{ru}, \\
\mathbf{C}_{pl} &\leq \mathbf{\Phi}_{k+1,k} \leq \mathbf{C}_{pu}, \\
\mathbf{C}_{hl} &\leq \mathbf{H}_k \leq \mathbf{C}_{hu}.
\end{aligned} \tag{23}$$

The constraints given in (23) involve a parameter vector α consisting of various filter parameters. The constant matrices $\mathbf{C}_{ql}, \mathbf{C}_{qu}, \mathbf{C}_{rl}, \mathbf{C}_{ru}, \mathbf{C}_{pl}, \mathbf{C}_{pu}, \mathbf{C}_{hl}$, and $\mathbf{C}_{hu}$ are possible

lower and upper bounds on the matrices $\mathbf{Q}_k, \mathbf{R}_k, \mathbf{\Phi}_{k+1,k}$, and $\mathbf{H}_k$.

To solve the constrained nonlinear program, a number of numerical methods can be used. The adaptation algorithm investigated here uses a RQP [31] approach to find the solution. Also, numerical derivatives are used because analytical expressions of the partial derivatives are quite complicated and hard to obtain.

A similar optimization set-up was suggested by Mehra [3]. Recognizing the difficulties that arise with the computation of the partial derivatives, Mehra makes a number of simplifying assumptions. In particular, he assumes that the system is time-invariant, and that the Kalman filter is in steady state. Then, he solves only for the steady state Kalman gain by using a suboptimal maximum likelihood adaptive filter (whose convergence is not guaranteed). Obviously, many systems of interest are time-varying, and require the entire history of the Kalman gain. Hence, there is a need to avoid such simplifying assumptions. This is accomplished by the proposed adaptation algorithm which solves directly for the unknown parameters by using the RQP method.

V. EXAMPLE I

In this section, we consider a simple tracking example to illustrate the performance of the adaptive Kalman filter bank. The target is traveling in a straight line at a constant velocity of 100 m/s. Thus the linearized equations of motion are given by

$$\mathbf{x}_{k+1} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \mathbf{w}_k \qquad (24)$$

where

$$\mathbf{x}_k = \begin{bmatrix} x_{1k} \\ x_{2k} \end{bmatrix} = \begin{bmatrix} \text{range} \\ \text{velocity} \end{bmatrix}$$

is the state vector of the target. It is assumed that range and velocity measurements are collected every $\Delta t = 1$ s. The measurement equation is then given by

$$\mathbf{z}_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}_k + \mathbf{v}_k \qquad (25)$$

where

$$\mathbf{z}_k = \begin{bmatrix} z_{1k} \\ z_{2k} \end{bmatrix} = \begin{bmatrix} \text{range measurement} \\ \text{velocity measurement} \end{bmatrix}$$

is the measurement vector. The $\mathbf{w}_k$ and $\mathbf{v}_k$ vectors are zero-mean, white sequences. The covariance matrices for the $\mathbf{w}_k$ and $\mathbf{v}_k$ vectors are given by

$$E\left[\mathbf{w}_k \mathbf{w}_i^T\right] = \mathbf{Q}_k \delta_{ki} = \begin{bmatrix} q_1 & 0 \\ 0 & q_2 \end{bmatrix} \delta_{ki}$$

and

$$E\left[\mathbf{v}_k \mathbf{v}_i^T\right] = \mathbf{R}_k \delta_{ki} = \begin{bmatrix} r_1 & 0 \\ 0 & r_2 \end{bmatrix} \delta_{ki}.$$

We also assume that process and measurement noise are uncorrelated.

The following experiments illustrate the capabilities of the proposed adaptive Kalman filter bank. In these simulations, a Kalman filter bank of size 5 is used. The unknown parameter vector consists of the system noise covariance matrix and the measurement noise covariance matrix. Hence, the parameter vector α is defined as

$$\alpha = \begin{bmatrix} q_1 & q_2 & r_1 & r_2 \end{bmatrix}^T. \qquad (26)$$

The initial conditions for the individual standard recursive Kalman filters in the bank are

$$\hat{\mathbf{x}}_0^- = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mathbf{P}_0^- = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}.$$

A. Experiment 1

We will first start by *examining the ability of the gating network to partition the input space.* This is done by modeling the third Kalman filter in the bank with the optimal parameter vector. Hence, the gating network should assign the optimal filter (i.e., KF 3) a weight close to unity as described earlier. The various values of the parameter vector for the individual filters are shown in Table I. Simulations were run for different values of the learning parameter η. The results for $\eta = 0.1$ and 1.0, shown in Fig. 6, illustrate the effect of the learning-rate parameter on the gating weight history. As expected, we find that increasing the value of the learning-rate parameter will result in a faster convergence to the optimal filter. However, we should note that the possibility of initially favoring a non-optimal filter increases if η is made too large.

B. Experiment 2

Next, we are interested in *illustrating the ability of the gating network to select the best performing filter when the optimal filter is not in the filter bank.* That is, the gating network should learn which filter is showing better performance. The optimal values of the parameter vector and the actual implemented parameters of the filter bank are shown in Table II. Again, simulations were run for different values of the learning parameter η. The results for $\eta = 0.1$ and 1.0, shown in Fig. 7, illustrate the effect of the learning-rate parameter on the gating weight history. We find that selection of the best performing filter is quite rapid. In all cases, the gating network selected the third Kalman filter in the bank since its parameter vector is "closest" to the optimal.

C. Experiment 3

We recall that the adaptive filter bank should have the *ability to switch among Kalman filters when the need arises.* To illustrate this capability of the filter bank, we ran a compound measurement sequence created by concatenating five sub-sequences of equal length, with the process and measurement noise statistics of the first sub-sequence best matching KF 1, that of the second sub-sequence best matching KF 2, and so on. That is, in each sub-sequence, the process and measurement noise statistics were changed. The values of the parameter vector used for these simulation are given in Table III. In these simulations, the learning-rate is $\eta = 1.0$. The results, shown in Fig. 8, illustrate the ability of the gating network to perform this task efficiently. The ability to process
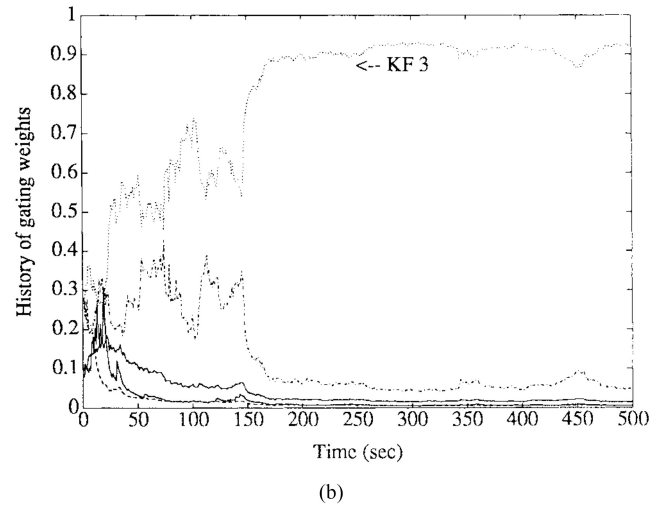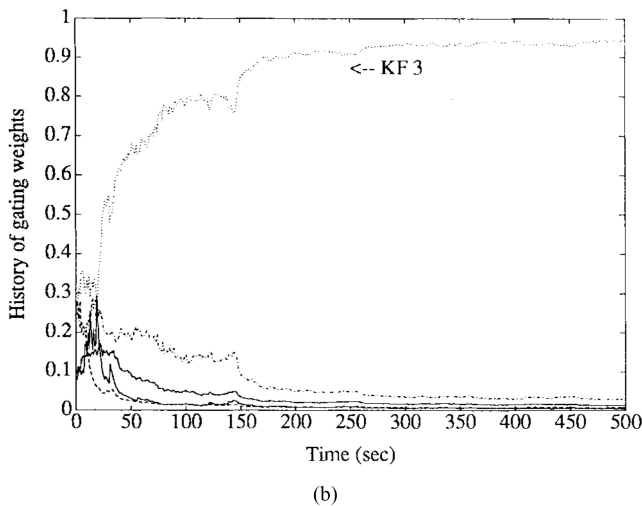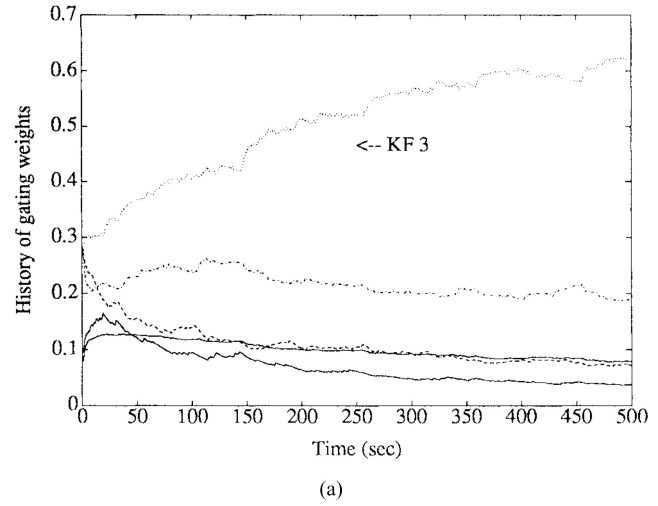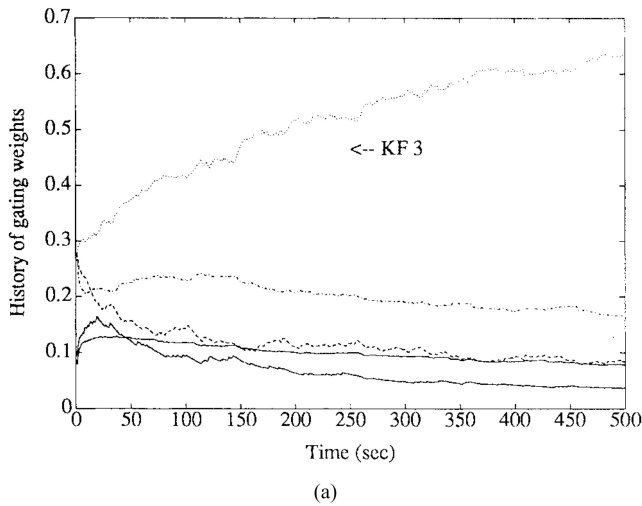
(a)

(b)

Fig. 6.   Gating weight history for two values of η with optimal Kalman filter (KF 3) included in the bank (Experiment 1). (a) Learning rate parameter $\eta = 0.1$ and (b) learning rate parameter $\eta = 1.0$.

(a)

(b)

Fig. 7.   Gating weight history for two values of η with optimal Kalman filter not included in the bank (Experiment 2). (a) Learning rate parameter $\eta = 0.1$ and (b) learning rate parameter $\eta = 1.0$.

TABLE I
PARAMETER VECTORS IN A FILTER BANK CONTAINING THE OPTIMAL (KF 3)

α	KF 1	KF 2	KF 3	KF 4	KF 5
q_1	1.0	9.0	16.0	25.0	5.0
q_2	1.0	9.0	16.0	25.0	20.0
r_1	1.0	25.0	16.0	9.0	25.0
r_2	1.0	25.0	16.0	9.0	10.0

TABLE II
PARAMETER VECTORS IN A FILTER BANK NOT CONTAINING THE OPTIMAL (KF*)

α	KF*	KF 1	KF 2	KF 3	KF 4	KF 5
q_1	16.0	1.0	9.0	12.0	25.0	5.0
q_2	16.0	1.0	9.0	12.0	25.0	20.0
r_1	16.0	1.0	25.0	20.0	9.0	25.0
r_2	16.0	1.0	25.0	20.0	9.0	10.0

a larger number of measurements is shown in Fig. 8(b). This demonstrates that the proposed approach is free from numerical underflow problems when processing large amounts of data.

For comparison purposes, the gating network is replaced with the Magill weighting scheme. We ran the 5 measurement sub-sequences of length 100 each, using the original weighting scheme. The obtained weight history is shown in Fig. 9(a). The scheme experiences a numerical underflow after processing 168 measurements. Fig. 9(b) is a plot of the weight history obtained using the gating network. To compare with the Magill scheme, the plot is shown only for the first 168 measurements. We observe that the need to switch to the second Kalman

filter is detected with the gating network which switches after measurement 103, as compared to the Magill scheme that does so only after measurement 110. Also, Fig. 9(a) and (b) reveals that the Magill algorithm was able to assign a unity weight to the appropriate filter before and after the switch in a relatively shorter time when compared to the gating network. To obtain a faster convergence of the weight with the gating network, the learning-rate parameter which controls the speed of convergence can be increased (as shown earlier in Experiments 1 and 2). For example, when $\eta = 7.0$, the gating network converges as fast as the Magill algorithm. Because of the underflow problem, comparison could not be made for sub-sequence of length 1000.
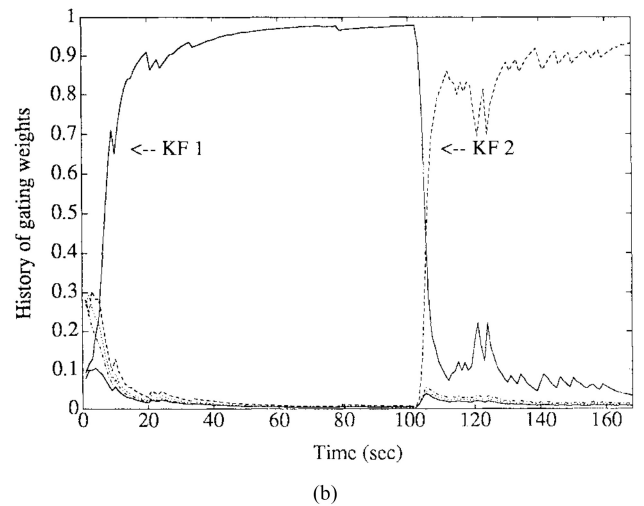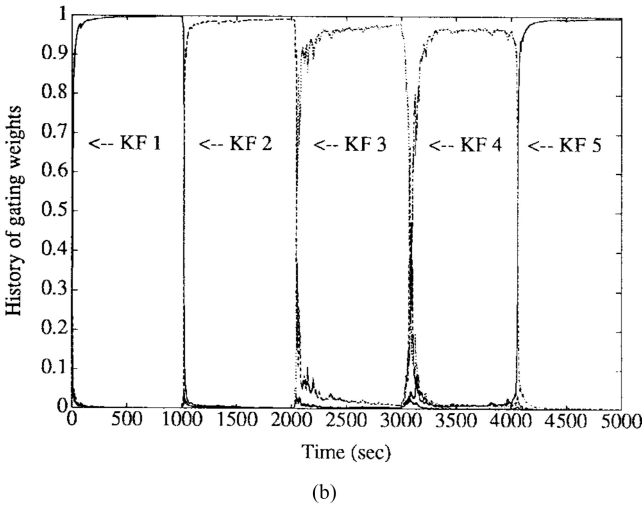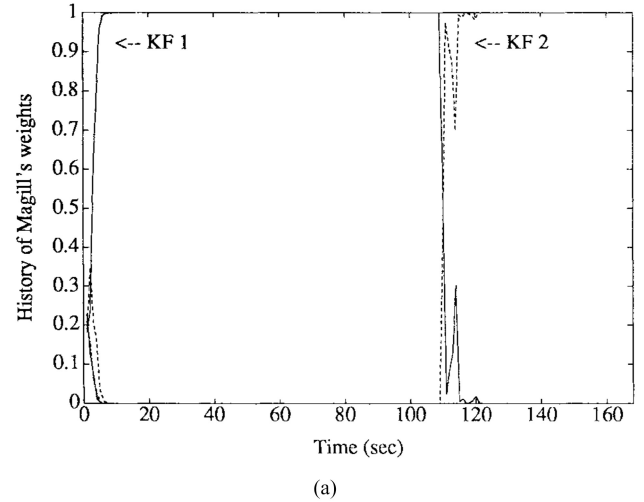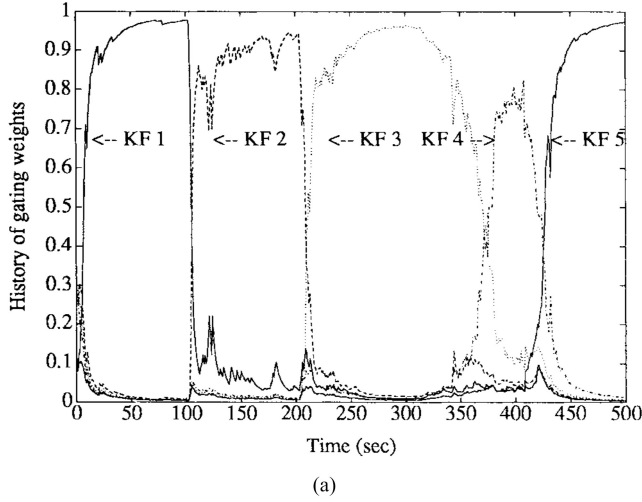
(a)

(b)

Fig. 8.   Switching capability of the adaptive Kalman filter bank (Experiment 3) with $\eta = 1.0$.

(a)

(b)

Fig. 9.   Comparison of approaches: (a) Magill's weighting scheme and (b) gating network with $\eta = 1.0$.

TABLE III
PARAMETER VECTORS FOR THE SWITCH EXPERIMENT

α	KF 1	KF 2	KF 3	KF 4	KF 5
q_1	1.0	10.0	10.0	5.0	1.0
q_2	1.0	25.0	10.0	25.0	1.0
r_1	1.0	1.0	16.0	2.0	2.0
r_2	1.0	1.0	16.0	25.0	50.0

D. Experiment 4

Finally, we will illustrate the *adaptation of a particular parameter vector of a filter in the bank by using a RQP search approach*. We start by initializing every Kalman filter in the bank with a different realization of the unknown parameter vector α. The learning-rate parameter used is $\eta = 1.0$. The initial bank parameters values are shown in Table IV. We will apply the adaptation algorithm periodically every 200 s. As indicated in Fig. 10, the gating network clearly prefers the first Kalman filter. This is an indication that the parameters of this filter are the best candidate. At $t = 200$ s, using the parameter vector of the best filter as an initial search point, the RQP search is initiated to look for the optimal parameter vector for the measurements processed in the last 200 s. For this first set

of measurements, the parameter vector obtained by solving the optimization problem (described earlier in Section IV) is

$$\alpha = [4.70 \quad 20.69 \quad 22.83 \quad 14.54]^T$$

which is close to the actual solution

$$\alpha^\star = [4.31 \quad 22.45 \quad 24.01 \quad 11.42]^T.$$

Since the obtained solution is quite different than the current parameter vector of the best filter (i.e., KF 1), filter adaptation is performed. In other words, the update would not be necessary if the obtained vector is close to the one currently used. Adaptation proceeds as follows:

1) Solution is provided to the Kalman filter which has the best performance;
2) Covariance matrices of all filters in the bank are re-initialized; and
3) Estimates of all filters are re-initialized with the estimate of the best Kalman filter prior to the update procedure.

Fig. 10 shows the gating weight history of the filter bank before and after the filter update procedure is performed. As expected, the first filter rapidly attains a high weight. As
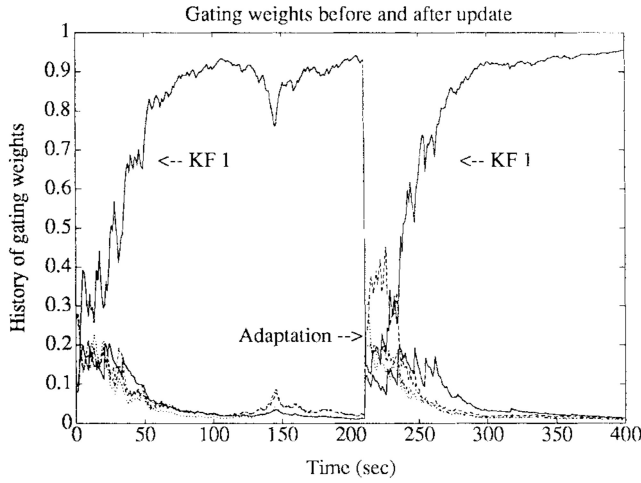
Fig. 10. Summary of gating weight history before and after adaptation (Experiment 4).

indicated earlier, only the parameters of the best filter are updated. The other filters are not changed to keep a diversity of models covering the unknown parameter search space. If a drastic change in parameters occurs then a switch to another filter will happen as shown in the previous experiment. Hence, it is desirable to keep the filter parameters models apart to rapidly respond to such changes. These models serve as initial search points in the parameter search space. In essence, the adaptation procedure is a periodic filter tuning procedure, wherein the parameter vector of the best filter is moved closer to the optimal one.

VI. EXAMPLE II

This section examines the problem of processing satellite radar altimetry measurements. One important use of space borne radar altimetry data is in the determination of short-wavelength features of the ocean and ice surface topography. Earth-orbiting satellites (carrying altimeters) have evolved into an important source of data for monitoring global oceans and ice glaciers. The data collected has allowed the study of the physical characteristics of the oceans [43]–[45]. This real world problem is an application where measurements are post-processed. It allows a realistic testing of the suggested mixture-of-experts approach.

The return signal strength of the radar altimeter, $s(t)$, can be modeled as

$$\dot{s}(t) = \beta(t)s(t) + w_1 \tag{27}$$

$$\dot{\beta}(t) = w_2$$

where w_1 and w_2 are zero-mean Gaussian processes satisfying

$$E[w_1(t)w_1(\tau)] = q_1(t)\delta(t - \tau),$$
$$E[w_2(t)w_2(\tau)] = q_2(t)\delta(t - \tau), \tag{28}$$
$$E[w_1(t)w_2(\tau)] = 0.$$

The measurement equation is given by

$$z_k = s_k + v_k \tag{29}$$

TABLE IV
PARAMETER VECTORS FOR INDIVIDUAL FILTERS BEFORE THE FIRST ADAPTATION

α	KF 1	KF 2	KF 3	KF 4	KF 5
q_1	20.0	0.0	10.0	5.0	50.0
q_2	20.0	0.0	10.0	5.0	50.0
r_1	20.0	5.0	1.0	5.0	50.0
r_2	20.0	5.0	1.0	5.0	50.0

where z_k is the observed value of s at time t_k. The measurement noise, v_k, is a zero-mean white sequence of known variance satisfying

$$E[v_k v_i] = \sigma^2 \delta_{ki}. \tag{30}$$

The filtering approach to this nonlinear system is to use the continuous-discrete extended Kalman filter (EKF) [46]. The derivation of the filter algorithm for this particular system model is detailed in [47]. For the satellite altimeter problem, filter performance is improved if the system noise parameters are adjusted to fit the data sample [48]. This is justified by the fact that $s(t)$ varies at a fixed location with time and geographically with latitude and longitude. Filter performance is verified by computing the root mean square (RMS) defined as

$$\text{RMS} = \frac{1}{p}\sum_{i=1}^{p}\left(z_i - \hat{s}_i^+\right)^2 \tag{31}$$

where p is the size of the measurement set and $\hat{s}_i^+$ is the state estimate of the power of reflected signal after incorporating measurement z_i (i.e., the updated state estimate). The unknown parameter vector α for this problem is defined as

$$\alpha = [q_1 \quad q_2]^T. \tag{32}$$

Currently, the system noise parameters for a particular data set are selected by a (generally) ad hoc procedure of making many computer simulations with various realizations of the unknown parameters until an acceptable RMS threshold is reached. The use of the mixture-of-experts approach to process satellite radar altimetry data sets represent a methodical way for adjusting the system noise parameters. In fact, a small number of EKF's each operating with a different set of the unknown parameter vector are considered. The learning gating network is used to identify the filter with the "best" parameter vector for a particular measurement set. Using the selection made by the gating network as a starting point, a RQP search is initiated to find a more suitable parameter vector for the processed measurement set. This recursive procedure can be repeated for every processed data set.

An EKF bank of size 5 was used to process real altimeter waveform data[1], shown in Fig. 11, the noise variance of which is $\sigma^2 = 0.25$. The 64 data points (known as gates) are collected at equal time intervals. They form a typical profile obtained over open ocean or large body of water. The initial conditions for the individual EKF's in the bank are

$$\begin{bmatrix} \hat{s}_0 \\ \hat{\beta}_0 \end{bmatrix} = \begin{bmatrix} 5 \\ -0.05 \end{bmatrix}, \quad \mathbf{P}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

[1] TOPEX data and initial filter conditions were provided by Dr. P. A. M. Abusali, Center for Space Research, University of Texas, Austin, TX 78712 USA.
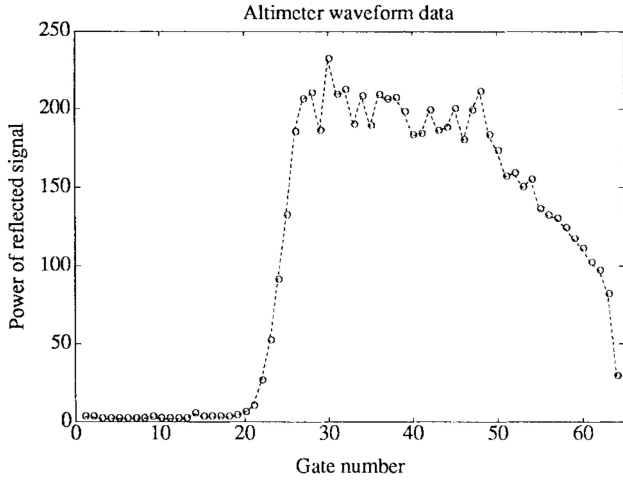
Fig. 11. Profile of real altimeter data over open ocean.

TABLE V
PARAMETER VECTORS FOR INDIVIDUAL FILTERS

α	EKF 1	EKF 2	EKF 3	EKF 4	EKF 5
q_1	10^{-2}	10^{-3}	10^{-1}	10^{-5}	10^{-3}
q_2	10^{-4}	10^{-5}	10^{-3}	10^{-3}	10^{-3}

A. Step 1

The initial filter parameters of the various EKF's are shown in Table V. The values of q_1 and q_2 are chosen to be less than unity to ensure that the uncertainty in our estimates (reflected by the covariance) will decrease as we process the measurement set. Simulations were run for different values of the learning-rate parameter η. The results for $\eta = 0.1$ and 1.0, shown in Fig. 12, suggest that the third EKF has the best parameter vector for the processed data set.

B. Step 2

Now that the mixture of EKF's regulated by the gating network has been used to identify the "best" parameter vector, the RQP search approach is used to obtain a more suitable solution. This is done by solving the optimization problem of Section IV for the processed measurement set. The obtained solution for the constrained nonlinear program using RQP is

$$\alpha^\star = [0.1985 \quad 0.0548]^T$$

which suggests a $q_1 \sim 10^{-1}$ and $q_2 \sim 10^{-2}$ for this particular altimeter waveform data.

C. Step 3

We examine the filter performance of the various models by computing the RMS defined earlier in (31). The results are shown in Table VI for the EKF's using the initial parameter sets and for an EKF using the parameter set found in step 2 (denoted EKF*). The obtained system noise parameters yield an RMS which is smaller by an order of magnitude when compared with the resulting RMS of the best model before initiating the search (i.e., EKF 3). This improvement in filter performance is significant. Also, a plot of $e_i = z_i - \hat{s}_i^+$ for these

(a)

(b)
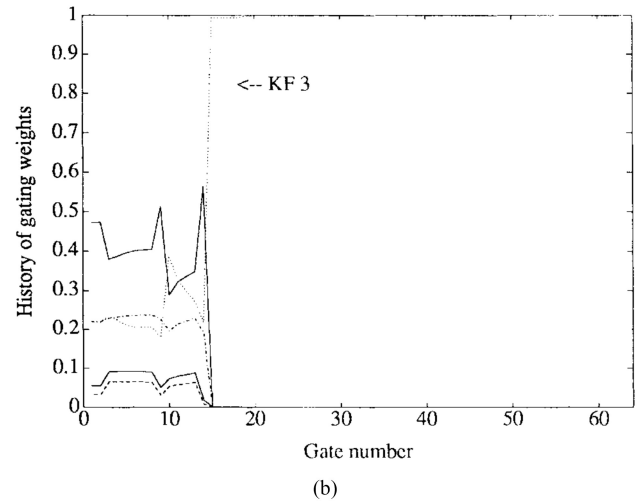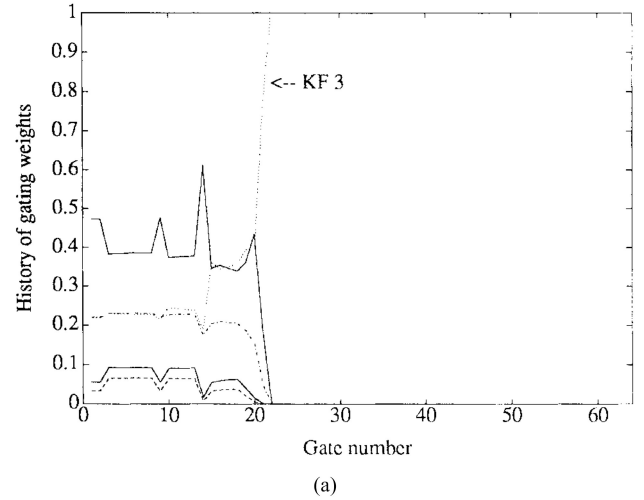
Fig. 12. History of gating weights for two values of η. (a) Learning rate parameter $\eta = 0.1$ and (b) learning rate parameter $\eta = 1.0$.

TABLE VI
PERFORMANCE OF VARIOUS FILTERS

	EKF 1	EKF 2	EKF 3	EKF 4	EKF 5	EKF*
RMS	4.2379	25.6593	0.2937	0.3923	0.3908	0.0101

two models is shown in Fig. 13 to illustrate the improvement in the estimation error.

VII. PARAMETER ADAPTATION WITH GENETIC ALGORITHMS

GA's are search algorithms based on the ideas and principles of biological evolution [32], [33]. These algorithms differ from the usual search algorithms in many ways. GA's consider many points in the search rather than a single point. This reduces the possibility of converging to a local extrema. Also, GA's do not work with the parameters themselves but rather with their coding (binary string), and use probabilistic transition rules in their search.

The properties of GA's lend themselves naturally to the modular structure of the filter bank. Generally, a larger number of filters are needed in the bank. This is due to the necessity of having a *population* of parameter vectors so that a GA use becomes possible. This approach to adaptation is not
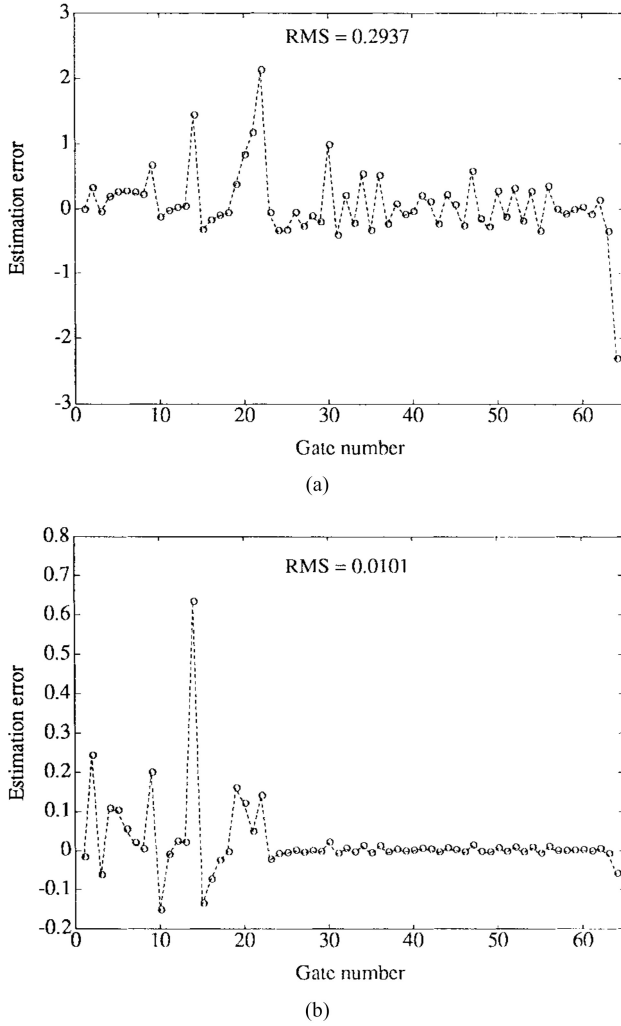
(a)

(b)

Fig. 13.   Comparison of estimation errors: (a) EKF 3 (best among current) and (b) EKF* (revised from RQP).

naturally implementable in real-time applications. Rather, it is a scenario which can be used for applications where the filter bank parameters need to be found after collection of the measurement set. The advantages of using a GA is the simplification of the adaptation procedure. In essence, a GA will only need the *fitness* of the current filter parameter models to start a search. This is readily available since it is just the gating weight provided by the network. Fig. 14 illustrates the operation of the adaptive filter bank with a GA in the feedback loop.

VIII. EXAMPLE III

We will consider the same illustrative example of Section V. We will also assume that the interest is to determine both the process noise covariance and measurement noise covariance matrices for the same set of 200 measurements of Experiment 4 in Section V. A Kalman filter bank of size 20 is used. We start by initializing every filter of the bank by *randomly* generating a binary string of length $l = 10$ for each component of the unknown parameter vector. The integer I represented by that string is then mapped to a real number M in the parameter search space which was assumed to be $[a, b] = [1, 50]$ for all
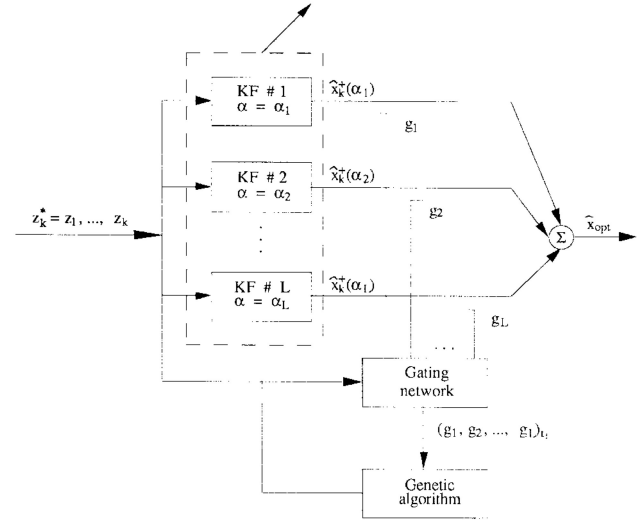

Fig. 14.   Adapting the Kalman filter bank with a genetic algorithm.

the components of α. The mapping function is given by

$$M = \frac{b - a}{2^l - 1} I + a. \tag{33}$$

The binary string representing a filter is a concatenation of the individual strings representing each of the unknown components of that filter. Since the parameter vector of this example contains four components, as shown in (26), the length of the string representing a particular filter parameters model has a length of $4l = 40$. The measurement set is processed by the filter bank. A gating network which has a unity learning-rate parameter is used to assess the performance of the various filter models. The bank adaptation procedure consists of supplying the weight values after processing the last measurement to the GA which has the binary coding of the current bank parameters. At this point, the GA will generate a new set of binary strings by 1) selecting two strings based on fitness, 2) choosing crossover positions (at the boundaries between the binary representations of each parameter value), and 3) using mutation after crossover. A common practice in GA's is to include the best performing string of the current generation in the new generation. This will serve two purposes. A good string would not be lost due to the probabilistic nature of GA's and convergence to the optimal solution is faster. The integer values represented by these strings are decoded and mapped to real numbers by using (33). This results in a new set of vectors $\alpha_1, \alpha_2, \dots, \alpha_{20}$. These are passed to the Kalman filter bank and the measurement set is processed again. This is repeated until the scheme converges or the maximum allowed number of iterations, 30, is reached. Convergence is checked by comparing the unknown parameter vector for all filters with a non-negligible weight. This is accomplished by checking the difference among the corresponding components for these filters. If the difference is within our specified tolerance, $\epsilon = 2$, then the iterations cease.

As seen in Fig. 15 (top left and top right), initially each filter is modeled around different value of the unknown parameter vector. These are generated randomly in the interval $[1, 50]$. The optimal components of $\alpha^\star$ are both indicated by "*" for (q_1, q_2) and (r_1, r_2) which represent the diagonal components
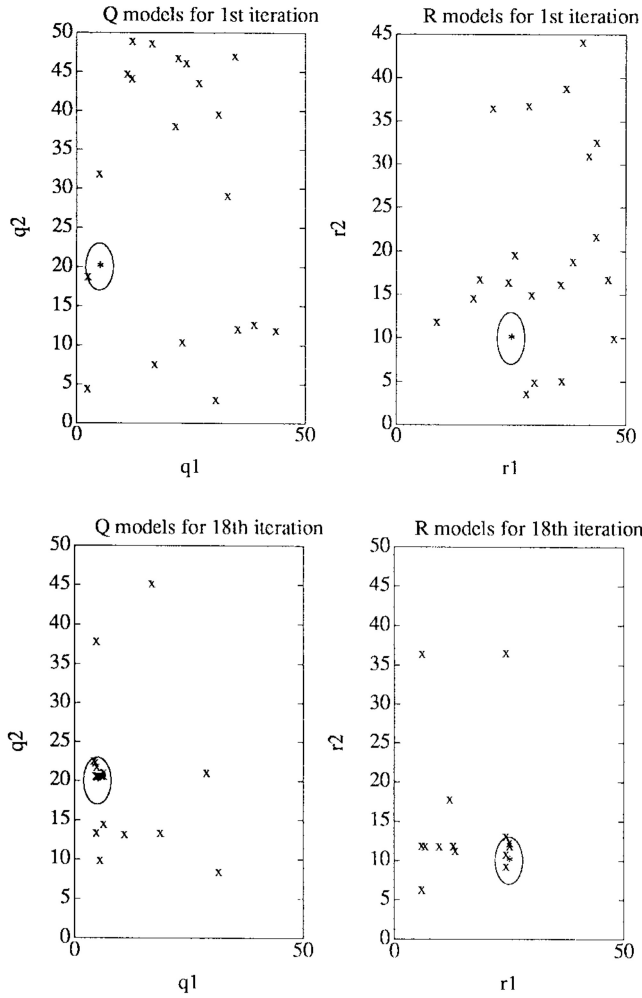
Fig. 15. Optimal Q and R (denoted *) with filters Q and R (denoted x).

TABLE VII
OPTIMAL AND BEST (1st AND 18th ITERATION) PARAMETER VECTOR

α	Optimal	Best 1st iter.	Best 18th iter.
q_1	4.31	2.34	6.22
q_2	22.45	18.58	20.45
r_1	24.01	8.66	24.18
r_2	11.42	11.71	11.66

of Q and R, respectively. Fig. 15 (b_1) and (b_2) shows the 18th iteration parameter values. The scheme has essentially converged. This is demonstrated by the clustering effect shown in Fig. 15 (b_1) and (b_2) where the plotted circles are centered around the optimal parameters. We note that some of the filters have a significantly different value for α. However, these filters are assigned a zero or negligible weight by the gating network. The best performing parameters in the 1st and 18th iteration are shown in Table VII with the optimal parameter values. The returned solution is consistent with the optimal.

IX. CONCLUSION

The modular adaptive Kalman filtering approach introduced in this paper has certain key advantages over conventional filter banks. The gating network adapts the weights given to individual filter outputs based on the observed sequence of measurements. This learning is unsupervised and the desired parameters need not be specified. The learning mechanism also does not assume that the optimal filter is in the bank. This feature, together with the periodic application of a search algorithm in the feedback loop, enables the system to provide good estimation accuracy even with small filter banks. The additive nature of the weight update procedure enables quicker response to changing environments and better numerical stability.

The thrust of the paper is on real-time applications. The RQP search algorithm can be applied concurrently while the filter bank and gating network are in operation. The GA alternative however is realistic only for non real-time applications where one may be willing to use much more compute power to obtain more accurate solutions.

The overall scheme is quite general and not restricted to using linear or extended Kalman filters as the individual experts. It can also be readily extended to hierarchical filter banks which involve multiple levels of gating [25]. The fundamental tradeoff between estimation accuracy and computational time involved in choosing i) the size of the filter bank and ii) when and how often to use the search algorithm, is problem dependent. Further experimentation on diverse data sets will help in better understanding such practical issues.

REFERENCES

[1] C. F. Price, "An analysis of the divergence problem in the Kalman filter," *IEEE Trans. Automat. Contr.*, vol. AC-13, pp. 699–702, Dec. 1968.
[2] R. J. Fitzgerald, "Divergence of the Kalman filter," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 736–747, Dec. 1971.
[3] R. K. Mehra, "Approaches to adaptive filtering," *IEEE Trans. Automat. Contr.*, vol. AC-17, pp. 693–698, Oct. 1972.
[4] ——, "On the identification of variances and adaptive Kalman filtering," *IEEE Trans. Automat. Contr.*, vol. AC-15, pp. 175–184, Apr. 1970.
[5] ——, "On-line identification of linear dynamic systems with applications to Kalman filtering," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 12–21, Feb. 1971.
[6] J. C. Shellenbarger, "Estimation of covariance parameters for an adaptive Kalman filter," in *Proc. Nat. Automat. Electron. Conf.*, 1966, vol. 22, pp. 698–702.
[7] A. P. Sage and G. W. Husa, "Adaptive filtering with unknown prior statistics," in *Proc. Joint. Automat. Contr. Conf.*, 1969, pp. 760–769.
[8] K. A. Myers and B. D. Tapley, "Adaptive sequential estimation with unknown statistics," *IEEE Trans. Automat. Contr.*, vol. AC-21, pp. 520–523, Aug. 1976.
[9] D. T. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Trans. Automat. Contr.*, vol. AC-10, pp. 434–439, Oct. 1965.
[10] C. G. Hilborn and D. G. Lainiotis, "Optimal estimation in the presence of unknown parameters," *IEEE Trans. Syst., Sci., Cybern.*, vol. SSC-5, pp. 38–43, Jan. 1969.
[11] F. L. Sims, D. G. Lainiotis, and D. T. Magill, "Recursive algorithm for the calculation of the adaptive Kalman filter weighting coefficients," *IEEE Trans. Automat. Contr.*, vol. AC-14, pp. 215–218, Apr. 1969.
[12] P. D. Abramson, "Simultaneous estimation of the state and noise statistics in linear systems," MIT Experimental Astronomy Lab., Cambridge, MA, Rep. TE-25, May 1968.
[13] R. K. Mehra, "Identification of stochastic linear dynamic systems using Kalman filter representation," *AIAA J.*, Oct. 1970.
[14] R. L. Kashyap, "Maximum likelihood identification of stochastic linear systems," *IEEE Trans. Automat. Contr.*, vol. AC-15, pp. 25–34, Feb. 1970.
[15] M. Athens, R. H. Whiting, and M. Gruber, "A suboptimal estimation algorithm with probabilistic editing for false measurements with appli-

cations to target tracking with wake phenomena," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 372–384, June 1977.

[16] C. B. Chang and M. Athens, "State estimation for discrete systems with switching parameters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-14, pp. 418–425, May 1978.

[17] J. K. Tugnait, "Comments on state estimation for discrete systems with switching parameters," *IEEE Aerosp. Electron. Syst.*, vol. AES-15, p. 464, May 1979.

[18] H. A. P. Blom, "An efficient filter for abruptly changing systems," in *Proc. 23rd IEEE Conf. Decision Contr.*, Dec. 1984, pp. 656–658.

[19] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Trans. Automat. Contr.*, vol. 33, pp. 780–783, Aug. 1988.

[20] D. Lerro and Y. Bar-Shalom, "Interacting multiple model tracking with target amplitude feature," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, pp. 494–509, Apr. 1993.

[21] X. R. Li and Y. Bar-Shalom, "Performance prediction of the interacting multiple model algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, pp. 755–771, July 1993.

[22] ——, "A recursive model approach to noise identification," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-30, pp. 671–684, July 1994.

[23] E. Daeipour and Y. Bar-Shalom, "An interacting multiple model approach for target tracking with glint noise," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 31, pp. 706–715, Apr. 1995.

[24] R. A. Jacobs and M. I. Jordan, "A competitive modular connectionist architecture," in *Advances in Neural Information Processing Systems 3*, R. P. Lippman *et al.*, Eds. San Mateo, CA: Morgan Kaufmann, 1991, pp. 767–773.

[25] ——, "Hierarchies of adaptive experts," *Advances in Neural Information Processing Systems 4*, J. E. Moody *et al.*, Eds. San Mateo, CA: Morgan Kaufmann, 1992, pp. 985–992, 1992.

[26] S. Haykin, *Neural Networks*. New York: Macmillan, 1994.

[27] R. G. Brown, "A new look at the Magill adaptive filter as a practical means of multiple hypothesis testing," *IEEE Trans. Circuits Syst.*, vol. CAS-30, pp. 765–768, Oct. 1983.

[28] A. A. Girgis and R. G. Brown, "Adaptive Kalman filtering in computer relaying: Fault classification using voltage models," *IEEE Trans. Power Apparatus Syst.*, vol. PAS-104, pp. 1168–1177, May 1985.

[29] R. G. Brown and P. Y. C. Hwang, "A Kalman filter approach to precision geodesy," *J. Inst. Navigation*, vol. 30, pp. 338–349, 1983–1984.

[30] G. L. Mealy, "Application of multiple model estimation to a recursive terrain height correlation system," *IEEE Trans. Automat. Contr.*, vol. AC-28, pp. 323–331, Mar. 1983.

[31] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations," in *Numerical Analysis: Proc. Biennial Dundee Conf. Numerical Analysis*, G. A. Watson, Ed. Berlin, Germany: Springer-Verlag, 1978, pp. 144–157.

[32] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.

[33] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[34] D. M. Etter, M. J. Hicks, and K. H. Cho, "Recursive adaptive filter design using an adaptive genetic algorithm," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, 1982, vol. 2, pp. 635–638.

[35] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME, J. Basic Eng., Ser. D*, vol. 83, pp. 95–108, Mar. 1961.

[36] R. E. Kalman, "New methods and results in linear prediction and filtering theory," in *Proc. Symp. Engineering Applications Random Function Theory Probability*. New York: Wiley, 1961.

[37] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering* New York: Wiley, 1992.

[38] M. I. Perrone and L. N. Cooper, "Learning from what's been learned: Supervised learning in multi-neural network systems," in *Proc. World Congr. Neural Networks*, 1993, vol. 3, pp. 354–357.

[39] R. A. Jacobs and M. I. Jordan, "Learning piecewise control strategies in a modular neural network architecture," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, pp. 337–345, Mar./Apr. 1993.

[40] K. Tumer and J. Ghosh, "A framework for estimating performance improvements in hybrid pattern classifiers," in *Proc. World Congr. Neural Networks*, INNS Press, 1994, vol. 3, pp. 220–225.

[41] J. S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationship to statistical pattern recognition," *Neuro-computing: Algorithms, Architectures, and Applications*, F. Fougelman-Soulie and J. Hérault, Eds. Berlin, Germany: Springer-Verlag, 1990, pp. 227–236.

[42] ——, "Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters,"

Advances in Neural Information Processing Systems 2, D. S. Touretzky, Ed. San Mateo, CA: Morgan Kaufmann, 1990, pp. 211–217.

[43] NASA, "Satellite altimetric measurements of the ocean: Report of the TOPEX science working group," Jet Propulsion Laboratory, Pasadena, CA, Rep. 400-111, 1981.

[44] R. H. Stewart, "Ocean wave measurement techniques," *Air Sea Interaction, Instruments, and Methods*, F. Dobson *et al.*, Eds. New York: Plenum, 1980, pp. 447–470.

[45] W. F. Townsend, J. T. McGoogan, and E. J. Walsh, "Satellite radar altimeters—Present and future oceanographic capabilities," *Oceanography From Space*, J. R. R. Gower, Ed. New York: Plenum, 1981, pp. 625–636.

[46] A. Gelb, *Applied Optimal Estimation*. Cambridge, MA: MIT Press, 1974.

[47] B. D. Tapley, B. E. Schutz, and P. A. M. Abusali, "A sequential method for filtering satellite altimeter measurements," *J. Geophysical Res.*, vol. 84, pp. 4061–4070, July 1979.

[48] P. Ugincius and J. Pinker, "Power spectral densities of Geos-3 Geoid heights and vertical deflections for four tracks in the North Atlantic," Naval Surface Weapons Center, Dahlgren, VA, Rep. TR-3561, Apr. 1977.

Wassim S. Chaer received the B.E. degree in computer and communication engineering from the American University of Beirut, Lebanon, in 1989, and the M.S. degree in systems and control engineering from the University of West Florida, Pensacola, in 1991. He is currently pursuing the Ph.D. degree at the University of Texas, Austin.

He is a Research Assistant with the Department of Aerospace Engineering and Engineering Mechanics at the University of Texas. His research interests include adaptive estimation and control, data fusion, optimization, neural networks, fuzzy logic and control, and search algorithms.

Robert H. Bishop (M'92) received the B.S. and M.S. degrees in aerospace engineering from Texas A&M University, College Station, in 1979 and 1980, respectively, and the Ph.D. degree in electrical and computer engineering from Rice University, Houston, TX, in 1990.

He was a Member of the Technical Staff at the Charles Stark Draper Laboratory, Cambridge, MA. In 1990, he joined the faculty at the University of Texas, Austin, where he is now Associate Professor and Assistant Department Chairman, Department of Aerospace Engineering and Engineering Mechanics. His research interests are in the area of guidance, navigation, and control.

Joydeep Ghosh received the B.Tech. degree from the Indian Institute of Technology, Kanpur, in 1983, and the M.S. and Ph.D. degrees in 1988 from the University of Southern California, Los Angeles.

He is currently an Associate Professor, Department of Electrical and Computer Engineering, University of Texas, Austin, where he holds the Endowed Engineering Foundation Fellowship. He directs the Laboratory for Artificial Neural Systems (LANS), where his research group is studying adaptive and learning systems. He has published six book chapters and more than 70 refereed papers.

Dr. Ghosh served as the General Chairman for the SPIE/SPSE Conference on Image Processing Architectures, Santa Clara, CA, in February 1990, Conference Co-Chair of Artificial Neural Networks in Engineering (ANNIE) in 1993–1995, and in the program committee of several conferences on neural networks and parallel processing. He received the 1992 Darlington Award given by the IEEE Circuits and Systems Society for the Best Paper in the areas of CAS/CAD, and also "best conference paper" citations for four papers on neural networks. He is an associate editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS and *Pattern Recognition*.